RULE-BASED DIGITAL VIDEO RECORDER

This application claims the benefit of U.S. Provisional Application No. 60/455,153, filed on March 14, 2003.

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to a method for storing video data on digital video recorders, and more particularly, to a method for prioritizing such data and overwriting previously recorded data according to predefined rules provided by the operator.

2. Description of the Prior Art

Video data is made up of numerous individual digital pictures, sometimes referred to as frames. These frames are still pictures, and must be run sequentially at high speeds to create the effect of motion within the video. The file size of each frame varies, depending upon its quality, dimensions, and other characteristics. The combination of these frames into a single video data file or stream results in the video data having a high volume of information, and consequently, a large file size. This file size increases with the duration, quality, dimensions, and other characteristics of the video. As a result, video data is generally larger in size than its traditional data file counterparts, even after compression routines are performed upon it. Therefore, applications involving the creation or maintenance of video data often require larger and/or more efficient storage means.

Digital video recorders store video data upon various storage media for future selectable review. These media are generally designed to hold anywhere from several hours to several weeks of video images. However, all media have finite capacities. Once such capacities are reached, the operator must insert new (empty) media into the video recorder, or recycle the current media for subsequent reuse. Depending upon the sheer volume of video data received

by the recorder, the former alternative may become cost-prohibitive within a short period of time.

Accordingly, reusing or recycling the media is commonly implemented. However, devices implementing such an alternative generally overwrite the existing video data in a sequential manner. This means that, once a storage medium is full, the recorder simply returns to the first sector of the medium and overwrites the existing video data from that point onward. Such a method does not take into consideration the potential importance or value of the data being overwritten – it simply overwrites the existing data in a sequential manner without regard to any priority the data may have. The result is that non-essential video may be retained, while essential video is deleted, simply because the essential events were recorded earlier in time.

It is therefore desirable to provide a method of reusing or recycling video data storage capacity once the storage medium has become full by overwriting the existing video data with new data such that the existing data is prioritized, so that existing low-priority data is overwritten instead of, or prior to, existing high-priority data. It is further desirable that the method be capable of receiving a real-time video data stream, and overwriting the existing data in real-time. It is further desirable that the priority of such existing data be redefinable in real-time.

SUMMARY OF THE INVENTION

The present invention provides methods for managing the overwriting of existing video data files or segments (discrete portions of video data) with new incoming video data by prioritizing all video and establishing rules for each level of priority. The method is based upon the classification or prioritization of the particular video segments into discrete categories, according to, for example, their sources, the events depicted within them, manual classification by the operator, or the like. The prioritization of the categories may be changed in real time by

the user, according to the value or importance of any particular video. Then rules are established, such as a minimum lifespan for each particular category, to determine how long the data should be kept. Information pertaining to such parameters may be stored within an index.

A first parameter categorizes the video segments according to such things as their source(s), the events depicted therein, the relationship of the segment to some event or alarm, or the like. Such sources or events generally fall into the following categories: video recorded on a constant basis; video recorded at scheduled dates and/or times; video recorded upon the occurrence of an anticipated and predefined event; video recorded upon the occurrence of an unanticipated (sometimes predefined) event; or video recorded upon changes in the environment. The category of a particular video segment may also be changed manually by the operator, or automatically upon the occurrence of certain events.

The operator may also prioritize the different categories, depending upon his or her particular requirements. Such priorities provide an initial determination as to which video segments are to be overwritten first by new video data. Specifically, and as described in greater detail herein, video segments having a lower priority are generally more likely to be overwritten than video segments having a higher priority.

Rules may be defined by the operator that provide a secondary determination as to when video segments of a particular category may be overwritten. The simplest form of such rules calls for comparison of the initial recording date of the particular video segment, to the system date at the time new video segments are being stored. As described in greater detail herein, a video segment may be overwritten only when its age equals or exceeds the minimum lifespan requirement for its particular category.

In use, the operator first establishes the priorities for each kind of video data that may be received (scheduled, alarm, archive, etc.). The user then establishes rules for the handling of

video data within each level of priority. For example, rules could be established for a minimum lifespan requirement for each category. Video data is then received by the digital video recorder, classified into one of the categories, and recorded upon the storage medium. The physical location of the video segment upon the medium, its priority level, and the date it was recorded, are placed in an index.

Once the storage medium is full, the present invention must overwrite the existing video data with new data. This is accomplished using a two-step filtering process. The present invention first filters the index to locate all existing video segments of the lowest priority category. It then filters the list of returned video segments by comparing the age of each segment with the minimum lifespan requirement for that particular category. Those segments younger than the minimum lifespan requirement are removed from the list so that they are saved. The remaining video segments may be overwritten. If only one segment remains after both filters are applied, it is overwritten by the new video data. If more than one video segment remains, the remaining segments are all compared against one another using the user-defined rules to determine which should be overwritten first. According to age-based rules, the oldest segment would then overwritten by the new data.

If there are no video segments within that particular category, or if all video segments in that category do not satisfy the rules (e.g., their ages do not equal or exceed the minimum lifespan requirement), the invention proceeds to the next higher priority category, and repeats the two-step filtering process upon the existing video segments of that higher category. Eventually, one or more suitable video segments will be located and overwritten by the new data. The present invention then repeats the process for each subsequent block of new data, beginning once again with the lowest priority category. Thus, according to the methods of the invention, video segments having the lowest priority are generally overwritten first by new

incoming video data, while higher priority files are maintained for longer periods of time, if not indefinitely.

The rules such as the minimum lifespan requirement for any category may be changed by the operator at any time. Such changes will affect all subsequent determinations under the second step of the two-step filtering process described herein, since those subsequent determinations will be evaluated using the new rule (e.g., the minimum lifespan requirement).

It is therefore a primary object of the present invention to provide a method for managing the use and reuse of data storage capacity once that storage capacity has become full, namely by overwriting the existing video data with new data according to a set of user-defined priorities and rules.

It is another primary object of the present invention that existing video data be prioritized by the operator, such that low-priority data is overwritten instead of, or prior to, high-priority data.

It is another object of the present invention that it be capable of receiving real-time video data streams from external sources, and overwriting, in real-time or substantially real-time, the existing video data upon the storage medium, according to an operator-defined data priority list and rules.

It is another object of the present invention that the priorities for the video data categories be adjustable in real-time, so that the adjustments have a real-time effect upon the preservation of the applicable video data segments.

Additional objects of the invention will be apparent from the detailed description and the claims herein.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a flowchart depicting general steps of the present invention.

Fig. 2 is an illustration of a typical index of the present invention.

Fig. 3 is an illustration of an index of the present invention, wherein a single block of video data has been recorded upon the storage medium.

Fig. 4 is an illustration of an index of the present invention, wherein the entire storage medium has been filled with video data.

Fig. 5 is an illustration of an index of the present invention, wherein one block of new video data is recorded upon the storage medium by overwriting one block of existing video data.

Fig. 6 is an illustration of the index of Fig. 5, wherein the index is aged by a period of two days.

Fig. 7 is an illustration of the index of the present invention, wherein one block of new video data is recorded upon the storage medium after a change in the minimum lifespan requirements.

DETAILED DESCRIPTION

Referring to the drawings wherein like reference characters designate like or corresponding parts throughout the various views, and referring particularly to Fig. 1, it is seen that the present invention is a method for storing and overwriting video data based upon certain predetermined priorities and rules provided by the operator. The invention is based upon the parameters of an operator-definable classification 10 of the video data into discrete categories, an operator-definable prioritization 11 of the categories, and operator-definable rules 12, such as a minimum lifespan requirement, for each priority.

The video data sources are categorized 10 by the operator on one or more of the following bases: its particular source (camera, location, scene, etc.); the events depicted therein (movement, etc.); or upon manual review by the operator. For example, video data received from constantly recording (twenty-four hours per day, seven days per week) cameras may be categorized as "Routine Recordings." Data received from cameras activated according to a predefined schedule, such as 8:00 a.m. to 5:00 p.m., Monday through Friday, may be categorized as "Scheduled Recordings." Videos of regular predefined events, such as movement across the camera's field of vision or the opening of a door, may be categorized as "Event Recordings." Videos of unexpected events, such as the breaking of a window, may be categorized as "Alarm Recordings." Finally, the operator may review video from any one of these categories and determine that it should be preserved; such video would be manually classified by the operator as "Archived Recordings." Other categories may be also be defined, according to the particular requirements of the operator.

The category classification(s) 10 of one or more video segments may be manually altered at any time by the operator. This feature is used when the operator wishes to affect the retention of one or more particular video segments. For example, the operator may determine, after reviewing a particular video segment, that it should be preserved for a longer period of time than otherwise provided for by its particular category. The operator may change the current segment category to one that provides a longer retention period, or define a new category entirely (such as "Archived Recordings"), with a higher priority, and assign the segment to that new category. The category may also be altered automatically upon the occurrence of a predefined event. For example, the occurrence of an alarm recording may cause the previous ten minutes of the scheduled recording to be promoted to an alarm category (having a higher priority) for longer preservation.

The user-defined categories are prioritized 11 according to the operator's particular criteria, based upon the operator's determination as to the necessity of preserving such video. As described in greater detail herein, such priorities will provide the first step of a two-step filtering process to determine which particular video segments will be erased to create available capacity upon the storage medium for new video segments.

Rules may also be established for each priority. For example, and without limitation, each priority may also be assigned a minimum lifespan requirement 12. Such a minimum lifespan requirement 12 would be used in the second step of the filtering process, and represents the minimum age of the video segment before it is subject to being selected for overwriting. As described in greater detail herein, such a requirement 12 prevents more recent video segments from being erased by subsequent data while older, and more suitable, video segments are available. Other rules may also be implemented to affect which video within a given category is to be overwritten first including, for example, and without limitation, the size of the video segment(s), the relationship of the video segment to other video segments, the type of data storage media upon which the video segment is stored, the particular data storage device used to store the video segment, and the like.

The rules such as the minimum lifespan requirement 12 of one or more priority categories may also be altered at any time. This feature is used when the operator wishes to affect the retention of all video segments within a particular priority category. Such alteration causes all subsequent determinations under the second step of the two-step filtering process to be evaluated under the new rule (e.g. changing the minimum lifespan requirement of priority 2 video from 3 days to 6 days).

Fig. 2 depicts a typical exemplary index 20 of the present invention. The index comprises entries for each video data segment recorded upon the storage media. Each entry

(one per video segment) comprises of the following information: the physical location 21, 22 (sectors, blocks and/or logical block addresses) of the video segment upon the storage medium; the priority category 23 of the segment, based upon the classification rule set provided by the operator; and other pertinent information 24 for user-defined rules such as the age or of the video segment, the media type upon which it is store, etc. The index may be of any form or combination of forms, such as arrays or linked lists.

In use, when new video data is recorded to the storage medium, the present invention first determines 13 if there are available empty blocks for recording the data. If empty blocks are available, the new data is recorded 19 upon those empty blocks. If there are no empty blocks available, or if there are an insufficient number of empty blocks available, the invention filters 14 the index to locate all existing video segments of the lowest priority category. The resulting list is then filtered 15 again, by applying the user-defined rules, such as comparing the "age" field of each eligible video segment entry to an established minimum lifespan requirement 12 for that particular category. Segments of that category, with ages equal to or exceeding the minimum lifespan requirements of that category, are made available for subsequent overwriting. If a plurality of eligible video segments remains after both filters are applied, all of them are compared 16 against one another on the basis of the applicable rule (e.g., their individual ages). The priority of the index entry for the oldest video segment among them is then set 18 to "empty." If only one entry remains after both filters are applied, its priority is set 18 to "empty" so that it is available to be overwritten.

In this example, if no video segments within the lowest priority category have ages equal to, or exceeding, the minimum lifespan requirement 12 for that priority category, the invention proceeds to the next higher category. The process described above is repeated, using the minimum lifespan requirement 12 for the new priority category. Such process may be

repeated several times until eligible storage capacity is found for the new video data in some priority category. The new video data is then recorded 19 at the physical location specified by the empty index entry, thereby overwriting the previous video segment stored at that physical location.

An alternative embodiment of the present invention further minimizes unused storage capacity by permitting several video segments to share a single block. The priority level of the shared block equals the highest priority level of the segments recorded upon the block. The age of the block may be equal to the oldest or youngest video segment recorded upon the block, depending upon operator preferences.

EXAMPLE OF THE PRESENT INVENTION IN ACTION

An example of the present invention is provided below. Assume that a store utilizes a system of four security cameras connected to a digital video recorder. The operator first assigns video output 10 from each camera to a particular category 11. In this example, the first camera is directed toward the rear alley exit, and provides a constant video stream of the exit (a routine recording). The second camera is directed toward the cashier's station, and is activated during regular store hours (a scheduled recording). The third camera is directed toward, and in communication with, the automatic doors of the store entrance, so that the camera is activated whenever the doors are opened (an event recording). The fourth camera is directed toward the manager's office, and in communication with a security keypad upon the door, so that it is activated only when the door is opened without first disabling the alarm system (an alarm recording).

In this example, assume that the operator has prioritized the different categories 11 of video data as follows: The alarm recording is assigned the highest priority (priority one),

followed by the event recording (priority two) and the scheduled recording (priority three). Finally, the routine recording is assigned the lowest priority (priority four). The operator also established a rule 12 assigning a minimum lifespan requirement for each category, as follows: seven days for the priority one alarm recordings; five days for the priority two event recordings; three days for the priority three scheduled recordings; and one day for the priority four routine recordings.

Fig. 2 depicts the index 20 of an empty storage medium. The storage medium used within this example is formatted to contain 100,000 logical block addresses (LBA), divided into ten blocks of 10,000 LBA each. It is to be understood, however, that thousands of blocks may exist upon an actual storage medium, depending upon the capacity of, and formatting method used upon, the medium. The "block" 21 and "logical block address" 22 fields represent the physical locations of the video segments upon the storage medium. This permits the video segments to be stored at any location upon the storage medium, allowing the present invention to incorporate methods of storage space optimization already known in the art. It also permits the video replay order to be sorted or rearranged, without having to actually move the video segments between physical locations upon the medium. The priority level of each video segment, as determined by the operator's previous ordering of categories, is recorded in the "priority" field 23. The age of each video segment, i.e. the number of days since the video was first recorded (obtained by comparing the current system date to the date the video was taken), is recorded in the "age" field 24.

Fig. 3 depicts the index 20 of the same storage medium after the first video segment 31 is recorded upon the medium. Here, the segment 31 is a routine recording (priority four) stored in block 0 (LBA 0-9999) of the medium. It is to be appreciated that the video segment does not have to be recorded at block 0; instead, because the physical location of each video segment is

recorded by the index, the segment may be recorded upon any empty block of the medium, so long as its physical location is reflected within the index 20. Because the segment was recently recorded, the "age" field 24 is 0.0 days.

Fig. 4 depicts the index 20 of the same storage medium after three days of operation. As indicated by the index 20 entries, the storage medium capacity has been reached. The video segments have been recorded in non-sequential order upon the medium, and comprise one alarm recording 41 (priority one, at block 4), one event recording 42 (priority two, at block 3), four scheduled recordings 43-46 (priority three, at blocks 1, 5, 6 and 7, respectively) and four routine recordings 31, 47-49 (priority four, at blocks 0, 2, 8 and 9, respectively).

Because the storage medium is now full, recording any new video data upon the medium will require one or more existing video segment(s) to be overwritten. Assume then, that a priority one video segment 51 (depicted in Fig. 5) is to be recorded upon the medium. The invention first verifies 13 that there are no empty blocks available. It then filters the index 20 to locate 14 all video segments within the lowest priority category (priority four). Here, the index returns a list indicating that blocks 0, 2, 8 and 9 (numerals 31, 47-49, respectively) contain video segments of priority four. The invention then filters 15 the list according to the one day minimum lifespan requirement for priority four segments. The video segment at block 8 (numeral 48) is 0.0 days old. It is therefore removed from the list, leaving the video segments at blocks 0, 2 and 9 (numerals 31, 47, 49, respectively). The index entries for these remaining video segments are then compared 16 against one another to determine which video segment among them is the oldest. Block 0 (numeral 31) is the oldest video segment. Therefore, its priority is set 18 to "empty" and it is overwritten 19 by the new video data 51, resulting in the index depicted in Fig. 5.

Further suppose that the operator subsequently changes the minimum lifespan requirements 12' for the various categories 10 to the following durations: ten days for the priority one alarm recordings; six days for the priority two event recordings; five days for the priority three scheduled recordings; and four and a half days for the priority four routine recordings.

Fig. 6 depicts the index of the same storage medium after two idle days, i.e. days in which no new video data is recorded upon the storage medium. The index 20 entries generally remain unchanged, except that the "age" field 24 of each entry is incremented to reflect the passage of time.

A priority two video segment 71 (depicted in Fig. 7) is to be recorded upon the medium. The invention first verifies 13 that there are no empty blocks available. It then filters the index 20 to locate 14 all video segments with the lowest priority category group (priority four). Here, the index 20 returns a list indicating that blocks 2, 8 and 9 (respectively, numerals 47-49) contain video segments of priority four. The invention then filters 15 the list according to the new minimum lifespan requirement 12' for the priority four segments. None of the priority four video segments equal or exceed the minimum lifespan requirements 12'. Therefore, none of the priority four segments will be overwritten. The invention then proceeds 17 to the priority three files 14'. The index returns a list indicating that blocks 1, 5, 6 and 7 (respectively, numerals 43-46) contain video segments of priority three. The invention then filters 15 the list according to the new minimum lifespan requirement for the priority three segments, leaving only the video segment at block 5 (numeral 44). The priority field of Block 5 is set 18 to "empty" and it is overwritten 19 by the new video data 71, resulting in the index depicted in Fig. 7.

As demonstrated by the foregoing example, video segments having lower priorities are overwritten first by the incoming video data, while higher priority files are maintained for

longer periods of time, if not indefinitely. The higher priority files are overwritten only if the lower priority files are too recent to be overwritten.

In another example, the rule processor makes the lowest priority space available for receiving new video data according to the following steps:

1. If empty block(s) are needed go to step 2, otherwise go to step 12.
2. Find the date of oldest block in the system with lowest priority.
3. If the date of lowest priority block is older than required by the rule for low priority blocks, then mark the block as "empty."
4. Repeat steps 2 and 3 until a date is found that is no longer older than required by rules.
5. If more empty block(s) are still needed, then repeat steps 2 through 4 using the next-lowest priority blocks.
6. If more empty blocks (s) are still needed, then repeat step 5 (steps 2 through 4) using the next lowest priority blocks, and so on until all priority levels have been searched.
7. If more empty block(s) are still needed then do the following:
8. Find the date of oldest block in the system with lowest priority.
9. Free the needed amount of lowest priority blocks by marking them as "empty."
10. If more empty block(s) are still needed, then repeat steps 8 and 9 using the next lowest priority blocks.
11. If more empty block(s) are still needed then repeat step 10 (steps 8 and 9) using the next lowest priority until enough blocks are empty to store the new video.
12. Store the new video in the empty blocks.

It is to be understood that variations and modifications of the present invention may be made without departing from the scope thereof. It is also to be understood that the present invention is not to be limited by the specific embodiments disclosed herein, but only in

accordance with the appended claims read in light of the foregoing specification.